BAB 3
ANALISIS DAN PERANCANGAN

3.1. Analisis Sistem

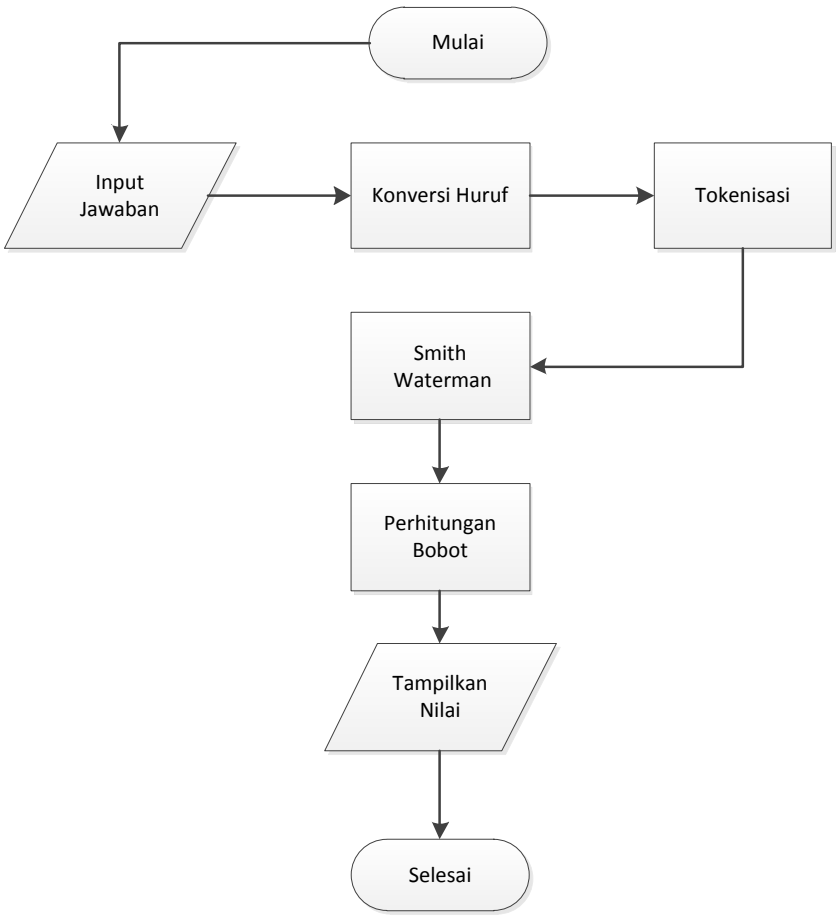
Analisis sistem (System Analysis) dapat didefinisikan sebagai penguraian dari suatu sistem informasi yang utuh ke dalam bagian-bagian komponennya dengan maksud untuk mengidentifikasikan dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya.

3.2. Analisis Masalah

Analisis masalah yang dalam penelitian ini adalah bagaimana menerapkan metode *smith waterman* sebagai penilaian pada jawaban essay secara otomatis dalam mata pelajaran bahasa Indonesia. Analisis masalah sangat penting dalam suatu penelitian, karena pada tahapan ini merupakan salah satu proses mengidentifikasi masalah-masalah yang berkaitan dengan bagaimana suatu sistem akan melakukan proses penilaian terhadap jawaban.

3.3. Analisis Metode

Metode *smith waterman* adalah metode yang dapat mengidentifikasi *local similarities* (penyejajaran sekuens). *Local similarities* yaitu proses penyusunan dua rangkaian/ susunan atau rentetan sehingga kemiripan antara dua sekuens tersebut akan terlihat. Berdasarkan fungsi proses penjajaran sekuens tersebut, algoritma ini dapat diterapkan kedalam bentuk pemograman komputer untuk mendeteksi kemiripan antara jawaban dan kunci jawaban.



Gambar 3.1 Alur Proses Smith Waterman

Berikut ini tahapan-tahapan beserta metode yang digunakan dalam aplikasi yang akan dibangun, mulai dari masukan hingga keluaran.

3.3.1 Input Jawaban

Pada tahap ini *user* mengerjakan soal-soal yang telah diinputkan. Setelah selesai mengerjakan soal, simpan jawaban kemudian akan diproses oleh tahap selanjutnya. Berikut ini salah satu contoh pertanyaan yang telah diinputkan :

Soal 1: Sebagai wartawan, tulisan-tulisan Ki Hajar Dewantara digemari para pemuda ketika itu, sebutkan alasan mengapa mereka menggagumi tulisan-tulisan Ki Hajar Dewantara?

Kunci Jawaban : Karena Ki Hajar Dewantara selalu menyampaikan tentang pentingnya persatuan dan kesatuan dalam berbangsa dan bernegara.

Berikut ini jawaban dari *user* tentang pertanyaan diatas, jawaban tersebut dapat dilihat pada tabel 3.1 sebagai berikut :

Tabel 3.1 Jawaban User

Soal	User	Jawaban User
1	Audiva	Karena ia selalu menyampaikan tentang persatuan dan kesatuan dalam berbangsa dan bernegara

3.3.2 Konversi Huruf

Konversi huruf merupakan salah satu proses yang terdapat pada algoritma smith waterman yang bertujuan untuk menyamakan tiap karakter dan merubah semua huruf dalam kalimat menjadi huruf kecil.

3.3.3 Proses Tokenisasi

Proses tokenisasi yaitu proses untuk membagi kalimat menjadi token-token atau bagian-bagian terkecil. Karakter spasi dan karakter lain berupa tanda baca pun dipisahkan. Adapun beberapa tanda baca yang dipisahkan adalah koma (,), titik (.), titik dua (:), garis miring (/), tanda tanya (?), dan tanda garis/pisah (-). Tanda baca dipisahkan guna memisahkan kata dari karakter tanda baca.

3.3.4 Proses Smith Waterman

Berikut adalah cara kerja dari algoritma *smith waterman* dalam menemukan kesamaan dari dua jawaban:

Jawaban = karena ia selalu menyampaikan tentang pentingnya persatuan dan kesatuan dalam berbangsa dan bernegara.

Kunci Jawaban = karena Ki Hajar Dewantara selalu menyampaikan pentingnya persatuan dan kesatuan dalam berbangsa dan bernegara.

Ket : Nilai *Match* = +2

Nilai *Mismatch* = -1

Nilai *Gap* = -1

Langkah awal dari algoritma *Smith-Waterman* adalah mendefinisikan nilai 0 pada matriks di titik (0,0), (i,0), dan (0,j). Kemudian dilakukan perhitungan pada titik (1,1). Apabila jawaban sama dengan jawaban j, maka akan digunakan nilai *match*. Sebaliknya, apabila jawaban dari kedua *sequence* berbeda, maka akan digunakan nilai *mismatch*.

$$H(1,1) = maks\begin{Bmatrix} H(i-1,j-1) + nilai\ match/mismatch \\ H(i-1,j) + nilai\ gap \\ H(i,j-1) + nilai\ gap \end{Bmatrix}$$

$$H(1,1) = maks\begin{Bmatrix} 0 \\ H(0,0) = H(0) + (2) = 2 \\ H(0,1) = H(0) + (-1) = -1 \\ H(1,0) = H(0) + (-1) = -1 \end{Bmatrix}$$

Bisa dilihat bahwa jawaban dari *i* dan dari *j* memiliki kesamaan, sehingga menggunakan nilai *match*. Nilai yang paling besar adalah 2, maka nilai pada titik (1,1) adalah 2. Perhitungan pada titik selanjutnya ditentukan dengan cara yang sama seperti pada titik (1,1).

	–	karena	ia	selalu	menyampaikan	pentingnya	persatuan	dan	kesatuan	dalam	berbangsa	dan	bernegara
–	0	0	0	0	0	0	0	0	0	0	0	0	0
karena	0												
ki	0												
hajar	0												
dewantara	0												
selalu	0												
menyampaikan	0												
tentang	0												
pentingnya	0												
persatuan	0												
dan	0												
kesatuan	0												
dalam	0												
berbangsa	0												
dan	0												
bernegara	0												

	–	karena	ia	selalu	menyampaikan	pentingnya	persatuan	dan	kesatuan	dalam	berbangsa	dan	bernegara
–	0	0	0	0	0	0	0	0	0	0	0	0	0
karena	0	2	1	0	-1	-1	-1	-1	-1	-1	-1	-1	-1
ki	0	1	1	0	-1	-2	-2	-2	-2	-2	-2	-2	-2
hajar	0	0	0	0	-1	-2	-3	-3	-3	-3	-3	-3	-3
dewantara	0	-1	-1	-1	-1	-2	-3	-4	-4	-4	-4	-4	-4
selalu	0	-1	-2	1	0	-1	-2	-3	-4	-5	-5	-5	-5
menyampaikan	0	-1	-2	0	3	2	1	0	-1	-2	-3	-4	-5
tentang	0	-1	-2	-1	2	2	1	0	-1	-2	-3	-4	-5
pentingnya	0	-1	-2	-2	1	4	3	2	1	0	-1	-2	-3
persatuan	0	-1	-2	-3	0	3	6	5	4	3	2	1	0
dan	0	-1	-2	-3	-1	2	5	8	7	6	5	4	3
kesatuan	0	-1	-2	-3	-2	1	4	7	10	9	8	7	6
dalam	0	-1	-2	-3	-3	0	3	6	10	12	11	10	9
berbangsa	0	-1	-2	-3	-4	-1	2	5	8	11	14	13	12
dan	0	-1	-2	-3	-4	-2	1	4	7	10	13	16	15
bernegara	0	-1	-2	-3	-4	-3	0	3	6	9	12	15	18

Gambar 3.2 Nilai Matriks dari Dua Buah Pasangan

Cara pengisian matriks :

H (1,2) maks = H (0,1) = H (0) + (-1) = -1
H (0,2) = H (0) + (-1) = -1
H (1,1) = H (2) + (-1) = 1

H (1,3) maks = H (0,2) = H (0) + (-1) = -1
H (0,3) = H (0) + (-1) = -1
H (1,2) = H (0) + (-1) = 0

H (1,4) maks = H (0,3) = H (0) + (-1) = -1
H (0,4) = H (0) + (-1) = -1
H (1,3) = H (0) + (-1) = -1

H (1,5) maks = H (0,4) = H (0) + (-1) = -1
H (0,5) = H (0) + (-1) = -1
H (1,4) = H (-1) + (-1) = -2

Berikut ini pseudcode nilai matriks dari dua buah pasangan, dapat dilihat pada tabel dibawah ini

Tabel 3.2 *Pseudcode* nilai matriks dari dua buah pasangan

```
length = conversi.length
int a = 0
array kunci = kunci jawaban
length = kunci.lenght
max_i = 0;
max_j = 0;
max_m =0;

for i = 1 i < length ; i++
for j = 1; j < length2; j++
    if conversi = kunci then
     a+match
   else
     if conversi = word sinonim then
     a+match
   else
     a+mismatch;

 end if

   create matriks
   m1 = m[i-1][j]+deletion
   m2 = m[i][j-1]+insertion

    If m[i][j]>max m
    max_h = m[i][j]
    max_i = i
    max_j = j
   end if
 end for

end for
```

Setelah dilakukan perhitungan dalam matriks, kemudian dilakukan *traceback* dari nilai tertinggi dalam matriks, yaitu dari titik (12,15). Hasil dari *traceback* yang dilakukan pada matriks adalah sebagai berikut.

	–	karena	ia	selalu	menyampaikan	pentingnya	persatuan	dan	kesatuan	dalam	berbangsa	dan	bernegara
–	0	0	0	0	0	0	0	0	0	0	0	0	0
karena	0	2	1	0	-1	-1	-1	-1	-1	-1	-1	-1	-1
ki	0	1	1	0	-1	-2	-2	-2	-2	-2	-2	-2	-2
hajar	0	0	0	0	-1	-2	-3	-3	-3	-3	-3	-3	-3
dewantara	0	-1	-1	-1	-1	-2	-3	-4	-4	-4	-4	-4	-4
selalu	0	-1	-2	1	0	-1	-2	-3	-4	-5	-5	-5	-5
menyampaikan	0	-1	-2	0	3	2	1	0	-1	-2	-3	-4	-5
tentang	0	-1	-2	-1	2	2	1	0	-1	-2	-3	-4	-5
pentingnya	0	-1	-2	-2	1	4	3	2	1	0	-1	-2	-3
persatuan	0	-1	-2	-3	0	3	6	5	4	3	2	1	0
dan	0	-1	-2	-3	-1	2	5	8	7	6	5	4	3
kesatuan	0	-1	-2	-3	-2	1	4	7	10	9	8	7	6
dalam	0	-1	-2	-3	-3	0	3	6	10	12	11	10	9
berbangsa	0	-1	-2	-3	-4	-1	2	5	8	11	14	13	12
dan	0	-1	-2	-3	-4	-2	1	4	7	10	13	16	15
bernegara	0	-1	-2	-3	-4	-3	0	3	6	9	12	15	18

Gambar 3.3 *Traceback* Matriks (12,15)

Memiliki skor : 0 + 2 - 1 + 2 + 1 + 2 + 2 + 2 + 2 +2 + 2 + 2= **18**

Dari *traceback* tersebut jika terdapat kata yang sama maka dihitung sebanyak 2 kali. Berikut ini *Local similarities* dari kedua jawaban tersebut adalah:

Kunci Jawaban : Karena ki hajar dewantara selalu menyampaikan tentang pentingnya persatuan dan kesatuan dalam berbangsa dan bernegara

Jawaban : Karena - - - - - - pentingnya persatuan dan kesatuan dalam berbangsa dan bernegara

Berikut ini *pseudcode traceback* pada matriks, dapat dilihat pada tabel 3.5

Tabel 3.3 *pseudcode traceback* pada matriks

```
1   Alignmet = array
2   value = m[max_i][max_j]
3   i = max_i-1
4   j = max_j-1
5   while value! = 0 and i!=0 and j!=0 do
6      s1 = mx[i]
7      s2 = my[j]
8      if s2! = null then
9           var_match <- null
10          if s1= s2 the
11          var_match <-1
12        end if
```

3.3.5 Perhitungan Bobot

Dari *Local Similarities*, bisa menghitung bobot kesamaan. Maka hasilnya adalah (9/15) x 100 = 60. Jadi, dapat disimpulkan dari kedua jawaban tersebut terdapat kecocokan 60%.

3.4. Sistem Kebutuhan Perangkat Lunak (SKPL)

Berdasarkan analisis masalah yang dibutuhkan oleh perangkat lunak, maka perangkat lunak harus memenuhi kriteria-kriteria berikut:

Tabel 3.4 SKPL

Kode	Keterangan
Fungsional	
SKPL – F1	Sistem dapat menampilkan soal
SKPL – F2	Sistem dapat melakukan proses login
SKPL – F3	Sistem dapat menampilkan hasil nilai
Non-Fungsional	
SKPL – NF1	Sistem mampu menerima inputan dari *mouse* dan *keyboard*
SKPL – NF2	Sistem mampu menampilkan pesan/*alert*

3.5. Analisis Kebutuhan Non-Fungsioanal

Analisis kebutuhan *non-fungsional* merupakan analisis yang dibutuhkan untuk menentukan spesifikasi kebutuhan sistem. Spesifikasi ini juga meliputi elemen atau komponen-komponen apa saja yang dibutuhkan untuk sistem yang akan dibangun sampai dengan sistem tersebut diimplementasikan. Analisis kebutuhan ini juga menentukan spesifikasi masukan yang diperlukan sistem, keluaran yang akan dihasilkan sistem dan proses yang dibutuhkan untuk mengolah masukan sehingga menghasilkan suatu keluaran yang diinginkan. Pada analisis kebutuhan sistem non fungsional ini dijelaskan analisis kebutuhan perangkat keras, analisis perangkat lunak, dan analisis pengguna.

3.5.1. Analisis Kebutuhan Perangkat Keras

Perangkat keras yang dibutuhkan untuk implementasi smith waterman pada aplikasi yang akan dibangun dapat dilihat pada tabel 3.5 sebagai berikut.

Tabel 3.5 Kebutuhan Perangkat Keras

Perangkat	Spesifikasi
Prossesor	Dual Core 2,2 G.hz
Memori	1 GB
Harddisk	80 GB
Monitor	LCD 14 Inch
Mouse	Standar
Keyboard	Standar

3.5.2. Analisis Kebutuhan Perangkat Lunak

Perangkat lunak yang dibutuhkan untuk implementasi smith waterman pada aplikasi yang akan dibuat. Adapun perangkat lunak yang dibutuhkan dapat dilihat pada Tabel 3.6.

Tabel 3.6 Spesifikasi Perangkat Lunak

No	Perangkat Lunak	Keterangan
1	Sistem Operasi	Windows 7
2	*Tools* Program	Browser Mozilla
3	Database	MySql Server
4	*Tools* Pengembang	Adobe Dreamweaver CS 5

3.5.3. Analisis Kebutuhan Pengguna

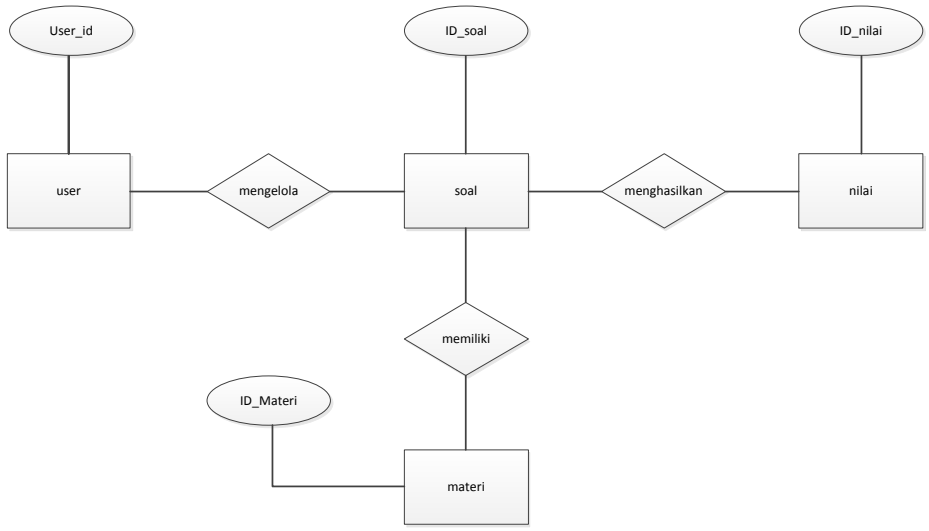
Analisis kebutuhan pengguna digunakan untuk mengetahui spesifikasi pengguna untuk dapat menjalankan aplikasi essay, seperti pada Tabel 3.7.

Tabel 3.7 Spesifikasi Kebutuhan Pengguna

Peran	***Users***
Tingkat Keterampilan	Dapat menggunakan komputer minimal mampu menggunakan beberapa aplikasi pendukung seperti Ms. Office, Mozilla Firefox.
Kondisi Fisik	Tidak buta total dan tidak buta warna.

3.6 Analisis Data

Analisis data pada sistem yang akan dibangun menggunakan *Entity Relation Diagram* (ERD). ERD adaah model data yang menggunakan notasi untuk menggambarkan data dalam konteks entitas dan hubungan yang di deskripsikan oleh data tersebut. ERD yang telah digunakan menggambarkan sistem yang terdiri dari hubungan antar entitas yang sedang berjalan. Untuk melihat keterhubungan antar entitas yang ada maka akan digambarkan pada gambar 3.4



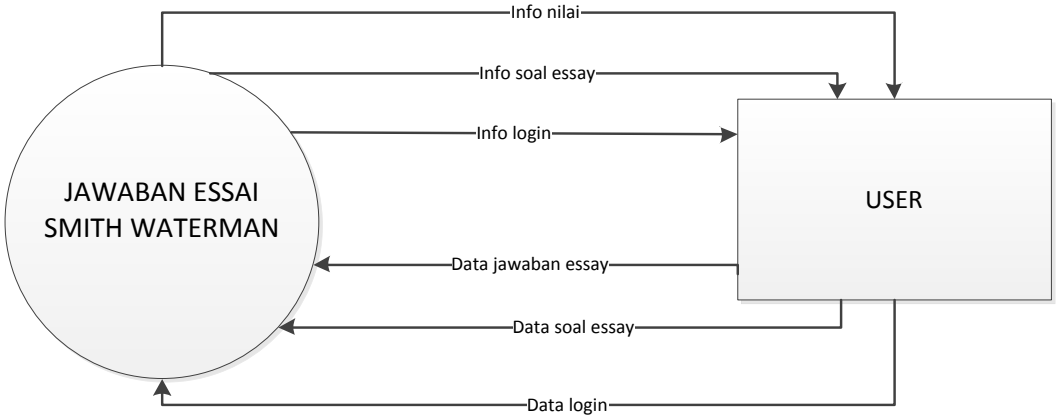
Gambar 3.4 Diagram ERD

3.7 Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional bertujuan untuk menganalisis proses yang akan diterapkan dalam sistem yang akan dibangun. Analisis kebutuhan fungsional pada aplikasi ini menggunakan Diagram Konteks, pembuatan DFD, dan Spesifikasi Proses.

3.7.1 Diagram Konteks

Diagram konteks adalah suatu diagram yang menggambarkan ruang lingkup media pembelajaran atau bagaimana media pembelajaran tersebut berinteraksi dengan lingkungannya. Sistem digambarkan dengan bulatan, sedangkan lingkungan diwakili oleh entitas luar yang digambarkan dengan persegi. Gambar 3.5 berikut ini merupakan diagram konteks Implementasi Metode Smith Waterman pada jawaban essay.



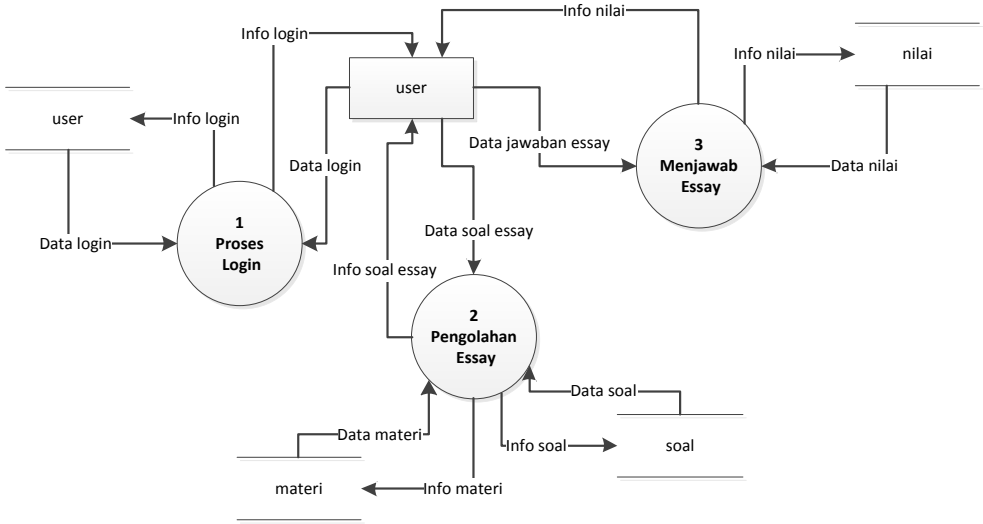
Gambar 3.5 Diagram Konteks Jawaban Essay Smith Waterman

3.7.2 Data Flow Diagram

DFD (Data Flow Diagram) merupakan sebuah teknik yang menggambarkan aliran data dan transformasi yang digunakan sebagai perjalanan data dari masukkan menuju keluaran. DFD dibuat jika pada Diagram Konteks masih terdapat proses yang mesti dijelaskan lebih rinci. Berikut ini adalah DFD pada aplikasi jawaban essay.

3.7.2.1 DFD Level 1

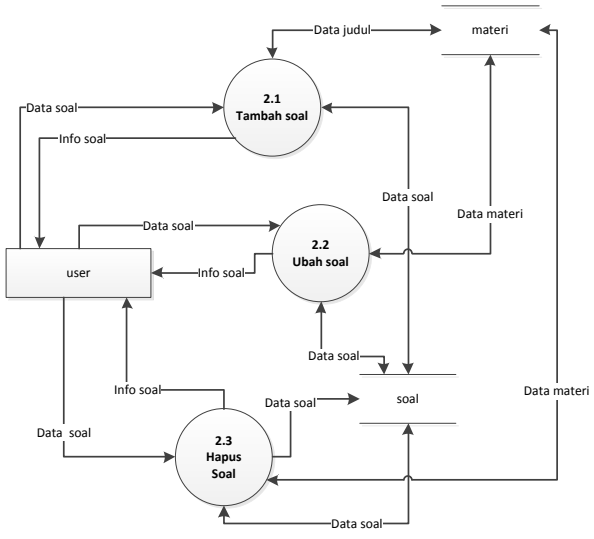
DFD Level 1 pada sistem terdapat 3 proses yaitu proses login, pengolahan essay, dan menjawab essay. DFD Level 1 dapat dilihat pada gambar 3.6.



Gambar 3.6 DFD Level 1 Jawaban Essay

3.7.2.2 DFD Level 2 Proses Pengolahan Essay

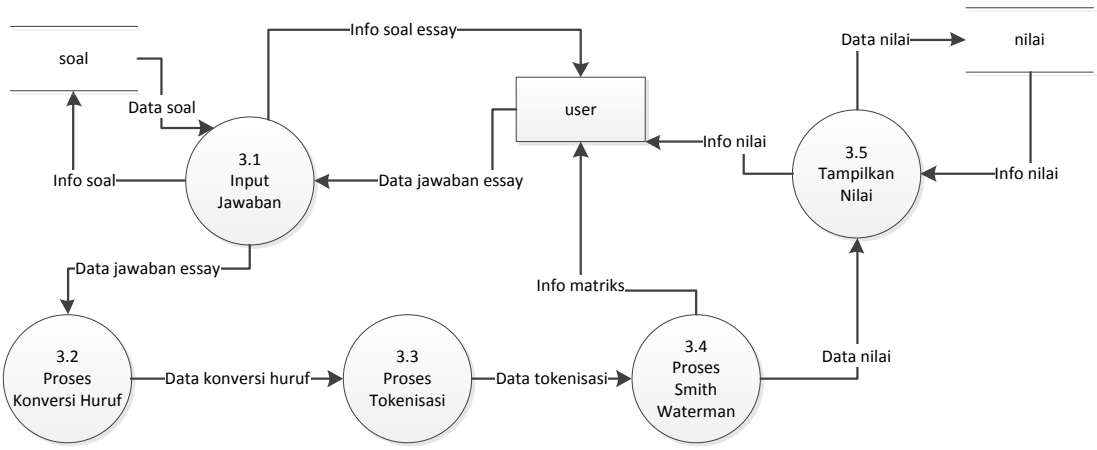
DFD Level 2 Proses 2 pada sistem terdapat 3 proses yaitu tambah soal essay, ubah soal essay dan hapus soal essay. DFD Level 2 Proses 2 dapat dilihat pada gambar 3.7.



Gambar 3.7 DFD Level 2 Proses 2 Pengolahan Essay

3.7.2.3 DFD Level 3 Proses Menjawab Essay

DFD Level 2 Proses 3 menjawab essay pada sistem terdapat 3 proses yaitu mengerjakan soal essay, proses smith waterman dan tampilkan nilai. DFD Level 2 Proses 3 dapat dilihat pada gambar 3.8.



Gambar 3.8 DFD Level 2 Proses 3 Menjawab Soal Essay

3.7.3 Spesifikasi Proses

Table 3.8 Spesifikasi Proses

No.	Proses	Keterangan
1	No. Proses	1
	Nama Proses	Proses Login
	Source	*User*
	Input	Data Login
	Output	Info Login
	Destination	*User*
	Logika Proses	1. *User* mengisi form *login*. 2. Sistem akan memeriksa *username* dan *password User*. 3. Apabila *username* dan *password* sesuai dengan yang tersimpan di database maka proses login berhasil, kemudian masuk ke beranda. 4. Apabila *username* dan *password* tidak sesuai dengan yang tersimpan di database maka proses login gagal.

No.	Proses	Keterangan
		5. *User* memasukan kembali *username* dan *password* sampai proses login berhasil.
2	No. Proses	2
	Nama Proses	Pengolahan Essay
	Source	*User*
	Input	Data essay
	Output	Info essay
	Destination	*User*
	Logika Proses	1. *User* memilih fungsionalitas olah essay. 2. Sistem akan menampilkan tampilan olah essay. 3. *User* dapat melihat data soal essay. 4. *User* dapat mencari soal essay. 5. *User* dapat menambah soal essay. 6. *User* dapat mengubah soal essay. 7. *User* dapat menghapus soal essay.
3	No. Proses	3
	Nama Proses	Menjawab Essay
	Source	*User*
	Input	Data jawaban essay
	Output	Info essay
	Destination	*User*
	Logika Proses	1. *User* memilih fungsionalitas lihat essay 2. Sistem akan menampilkan halaman menjawab essay. 3. *User* memilih jawab essay. 4. *User* melihat nilai.
4	No. Proses	2.1
	Nama Proses	Tambah soal
	Source	*User*
	Input	Data soal
	Output	Info soal
	Destination	*User*
	Logika Proses	1. *User* memimilih fungsional olah halaman menu soal essay. 2. Sistem akan menampilkan form data menu essay. 3. *User* menambahkan soal lalu simpan.

No.	Proses	Keterangan
		4. Sistem akan menyimpan data soal kedalam database.
5	No. Proses	2.2
	Nama Proses	Ubah Soal
	Source (sumber)	*User*
	Input	Data soal essay
	Output	Info soal essay
	Destination	*User*
	Logika Proses	1. *User* memilih fungsionalitas olah halaman menu soal essay. 2. Sistem menampilkan form data soal essay. 3. *User* mencari data soal essay yang akan diubah. 4. *User* mengubah data soal essay. 5. Jika menekan tombol ubah maka data soal essay berubah.
6	No. Proses	2.3
	Nama Proses	Hapus Soal
	Source	*User*
	Input	data soal essay
	Output	Info soal essay
	Destination	*User*
	Logika Proses	1. *User* memilih fungsionalitas olah halaman menu soal essay 2. Sistem menampilkan form data soal essay. 3. *User* mencari data soal essay yang akan dihapus. 4. *User* menghapus data soal essay. 5. Jika menekan tombol hapus maka data soal essay terhapus.
7	No. Proses	3.1
	Nama Proses	Input Jawaban
	Source	*User*
	Input	Data jawaban essay
	Output	Info soal essay
	Destination	*User*
	Logika Proses	1. *User* memilih fungsionalitas halaman menjawab soal essay. 2. Sistem akan menampilkan halaman menjawab soal essay. 3. *User* dapat melihat soal essay. 4. *User* menginputkan jawaban.

No.	Proses	Keterangan
8	No Proses	3.2
	Nama Proses	Proses Konversi Huruf
	Source	*User*
	Input	Data Jawaban Essay
	Output	Hasil Konversi Huruf
	Destination	*User*
	Logika Proses	1. Sistem mengecek jawaban soal essay. 2. Sistem akan merubah jawaban menjadi huruf kecil.
9	No. Proses	3.3
	Nama Proses	Proses Tokenisasi
	Source	*User*
	Input	Data Konversi Huruf
	Output	Hasil tokenisasi
	Destination	*User*
	Logika Proses	1. Sistem akan mengecek hasil konversi huruf. 2. Sistem akan menghapus jika terdapat tanda baca seperti (. “, -)
10	No. Proses	3.4
	Nama Proses	Proses Smith Waterman
	Source	*User*
	Input	Data Jawaban Essay
	Output	Info Matriks
	Destination	*User*
	Logika Proses	1. Sistem mengecek hasil konversi huruf dan tokenisasi. 2. Sistem akan menampilkan nilai matriks.
11	No. Proses	3.5
	Nama Proses	Tampilkan Nilai
	Source	*User*
	Input	Data Nilai
	Output	Info Nilai
	Destination	*User*
	Logika Proses	1. Sistem menampilkan nilai

3.7.4 Kamus Data

Kamus data merupakan sebuah daftar yang tersusun dari elemen data yang berhubungan dengan sistem. Kamus data dibuat berdasarkan arus data yang ada di data flow diagram. Berikut kamus data dapat dilihat pada tabel 3.9

Table 3.9 Kamus Data

No.	Proses	Keterangan
1	Nama	Data login
	Where used / how used	Proses 1 Login
	Deskripsi	Validasi *username* dan *password*
	Struktur data	USER_ID + USERNAME+ PASSWORD + NAMA
	USER_ID *USERNAME* *PASSWORD* nama	{A-Z | a-z | 0-9 | ‘ | - | . | , | |} {A-Z | a-z | 0-9 | ‘ | - | . | , | |} {A-Z | a-z} {A-Z | a-z | 0-9 | ‘ | - | . | , | |}
2	Nama	Data soal Essay
	Where used / how used	1. Proses 2 Pengolahan Essay 2. Proses 2.1 Tambah Soal Essay 3. Proses 2.2 Ubah Soal Essay 4. Proses 2.3 Hapus Soal Essay
	Deskripsi	Data ini digunakan untuk mengolah data soal essay.
	Struktur data	ID + ID_Materi + SOAL + JAWABAN
	ID ID_Materi SOAL JAWABAN	{0-9} {0-9} {A-Z | a-z | 0-9 | ‘ | - | . | , | |} {A-Z | a-z | 0-9 | ‘ | - | . | , | |}
3	Nama	Data Jawaban Essay
	Where used / how used	1. Proses 3 Menjawab Essay 2. Proses 3.1 mengerjakan soal essay 3. Proses 3.2 melihat nilai
	Deskripsi	Data ini digunakan untuk menjawab soal essay dan melihat nilai.
	Struktur data	ID + ID_Materi + NIM + NILAI + ID_SOAL + SIGN + JAWABAN_SISWA
	ID ID_Materi NILAI ID_SOAL	{0-9} {0-9} {0-9 | ‘ | - | . | , | |} {0-9}

No.	Proses	Keterangan
	SIGN JAWABAN_SISWA	{A-Z | a-z | 0-9 | ‘ | - | . | , | |} {A-Z | a-z | 0-9 | ‘ | - | . | , | |}

3.8 Perancangan Sistem

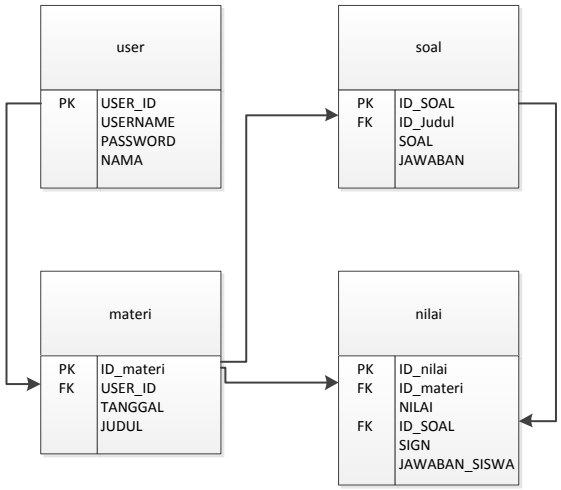
Perancangan sistem merupakan penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah kedalam satu kesatuan yang utuh dan berfungsi.

3.8.1 Perancangan Basis Data

Perancangan data terdiri dari skema dan struktur tabel yang terdapat pada ERD dari aplikasi jawaban essay.

3.8.1.1 Skema Relasi

Relasi antar tabel merupakan gabungan antar file yang mempunyai kunci utama yang sama, sehingga file-file tersebut menjadi satu kesatuan yang dihubungkan oleh field kunci (*primary key*). Pada proses ini elemen-elemen data dikelompokan menjadi satu file database beserta entitas dan hubungannya. Skema relasi aplikasi jawaban essay dapat dilihat pada gambar 3.9.



Gambar 3.9 Skema Relasi

3.8.2 Struktur Tabel

Tabel-tabel yang terdapat dalam basis data yang digunakan dalam sistem yang akan dibangun adalah sebagai berikut:

Tabel 3.10 Struktur Tabel User

Nama Field	Tipe Data	Key	Keterangan
USER_ID	Int (11)	PK	*Primary Key*
USERNAME	Varchar (100)		
PASSWORD	Varchar (32)		
NAMA	Varchar (150)		

Tabel 3.11 Struktur Tabel Materi

Nama Field	Tipe Data	Key	Keterangan
ID_Materi	Int (11)	PK	*Primary Key*
USER_ID	Varchar (30)	FK	*Foreign Key* dari tabel *user*
TANGGAL	timestamp		
JUDUL	Varchar (200)		

Tabel 3.12 Struktur Tabel Soal

Nama Field	Tipe Data	Key	Keterangan
ID_SOAL	Int (11)	PK	*Primary Key*
ID_Materi	Varchar (30)	FK	*Foreign Key* dari tabel materi
SOAL	Text		
KUNCI_JAWABAN	Text		

Tabel 3.13 Struktur Tabel Nilai

Nama Field	Tipe Data	Key	Keterangan
ID_NILAI	Int (11)	PK	*Primary Key*
ID_Materi	Int (11)	FK	*Foreign key* dari tabel essai

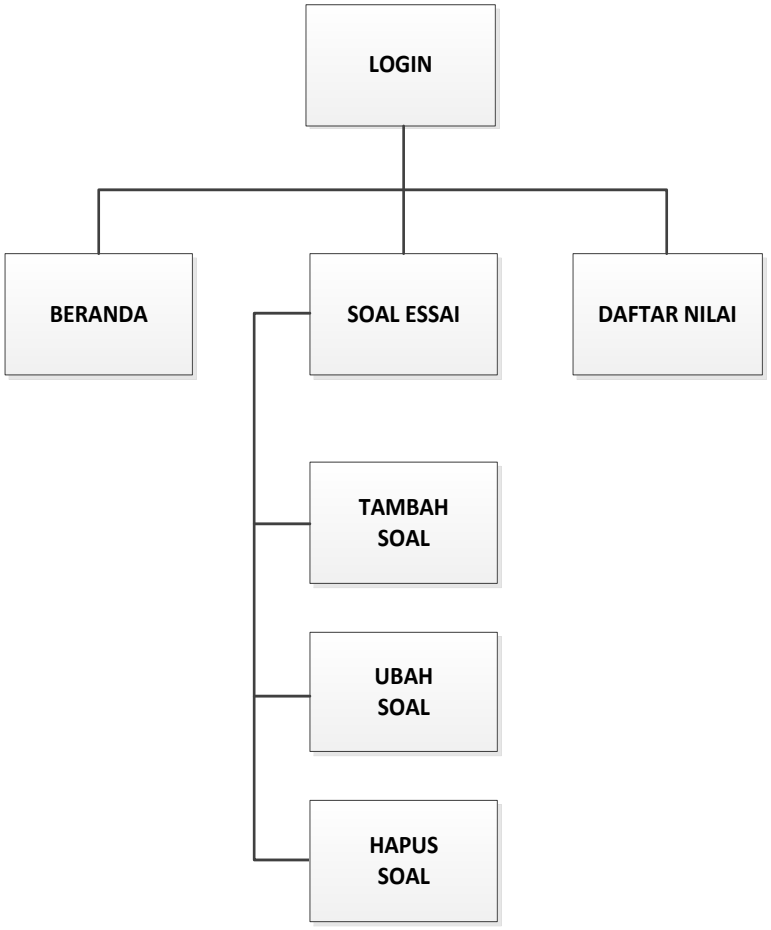
NILAI	Double		
ID_SOAL	Int (11)	FK	*Foreign key* dari tabel soal
SIGN	Text		
JAWABAN	Text		

Tabel 3.14 Struktur Tabel Sinonim

Nama Field	Tipe Data	Key	Keterangan
ID_SINONIM	Int (11)	PK	*Primary Key*
WORD	Varchar (50)		
WORD_SINONIM	Varchar (50)		

3.8.3 Perancangan Struktur Menu

Perancangan struktur menu berisikan menu dan submenu yang berfungsi memudahkan pengguna didalam menggunakan sistem. Berikut ini gambaran mengenai struktur menu aplikasi jawaban essay dapat dilihat pada gambar 3.10.

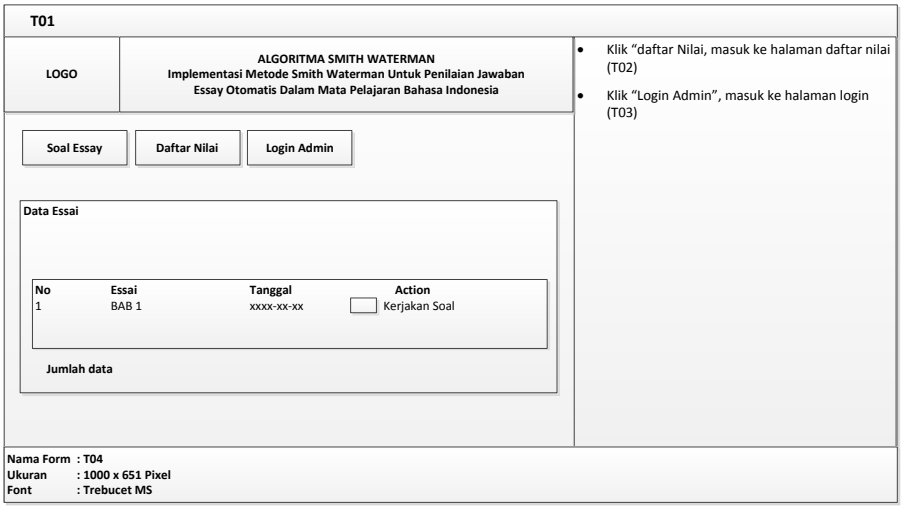


Gambar 3.10 Struktur Menu

3.8.4 Perancangan Antarmuka

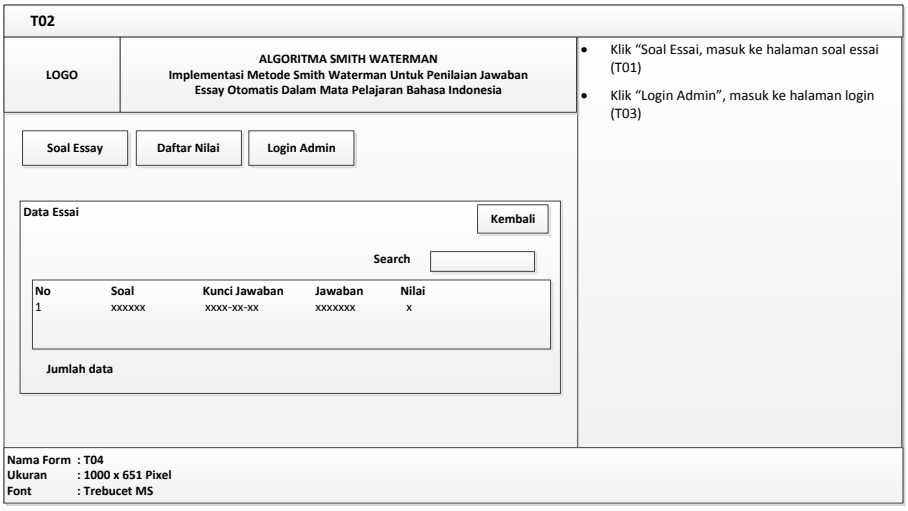
Tahap perancangan antarmuka dilakukan untuk merancang antarmuka agar dapat digunakan oleh pengguna. Perancangan antarmuka pada aplikasi jawaban essay adalah sebagai berikut.

1. Halaman Soal Essay (T01)



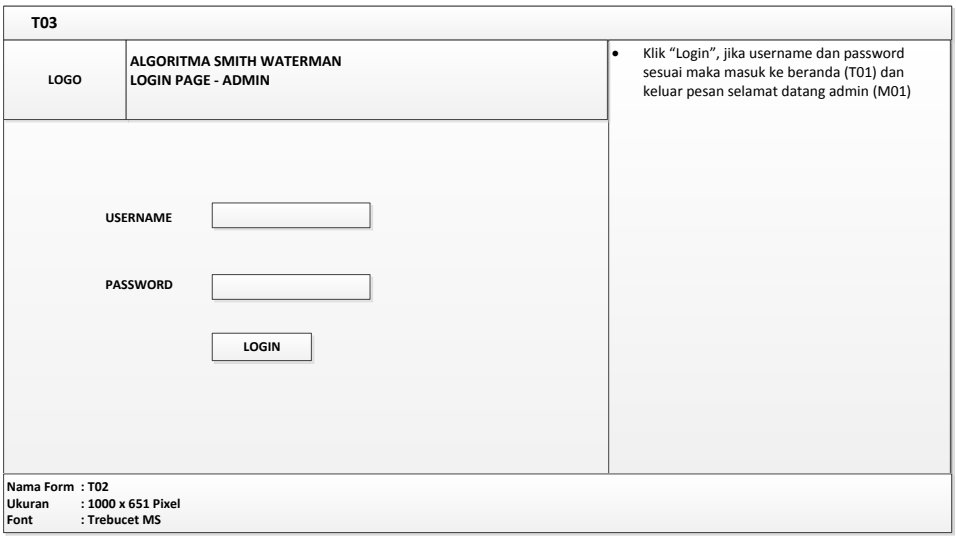
Gambar 3.11 Antarmuka Soal Essay

2. Halaman Daftar Nilai (T02)



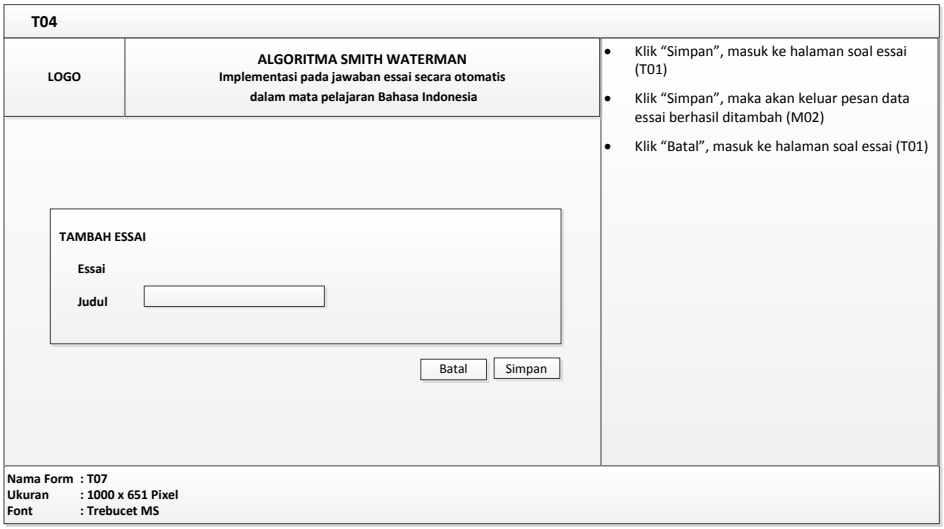
Gambar 3.12 Antarmuka Daftar Nilai

3. Halaman Login



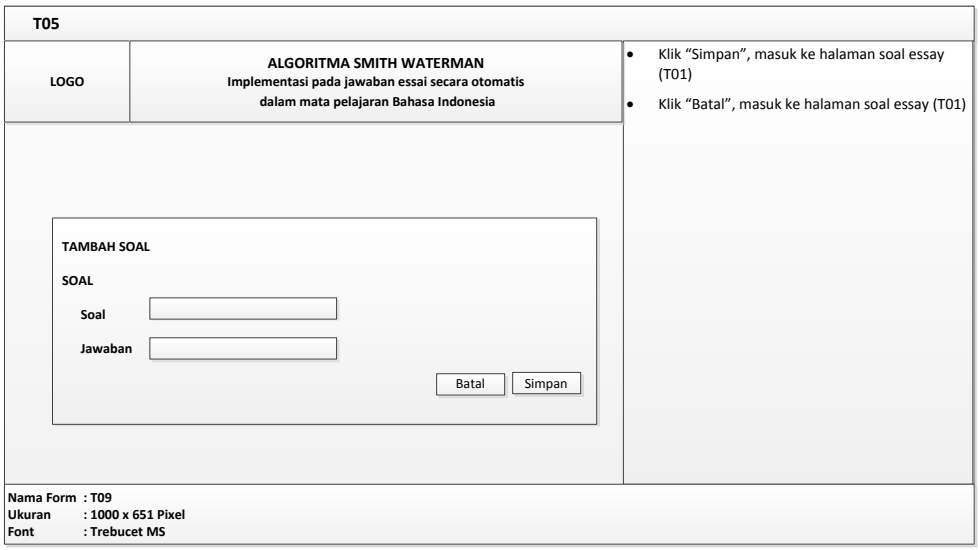
Gambar 3.13 Antarmuka Login

4. Halaman Tambah Essai



Gambar 3.14 Antarmuka Tambah Essay

5. Halaman Tambah Soal



Gambar 3.15 Antarmuka Tambah Soal

3.8.4.1 Perancangan Pesan

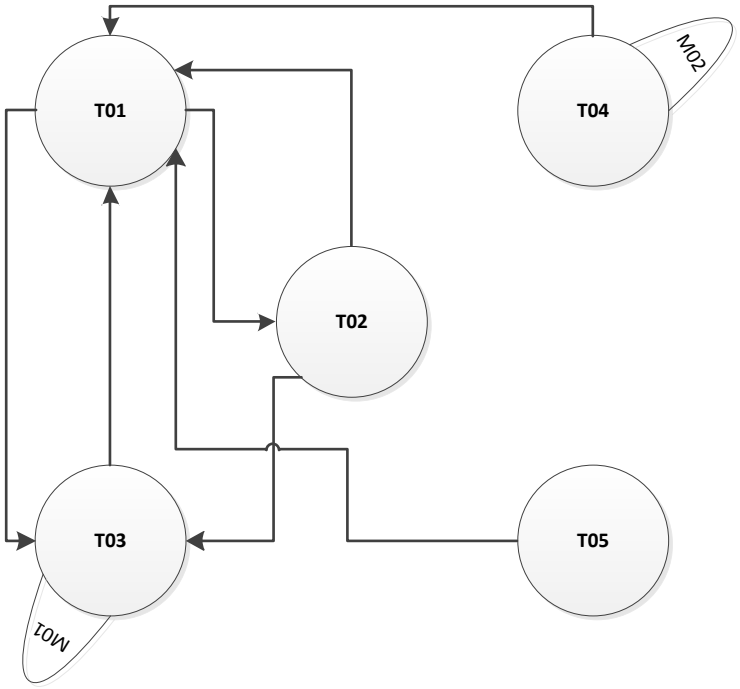
Perancangan pesan merupakan tampilan dari suatu perangkat lunak yang berfungsi untuk menyampaikan pesan atau informasi kepada pengguna agar perangkat lunak lebih interaktif. Perancangan pesan dapat dilihat pada gambar 3.16



Gambar 3.16 Perancangan Pesan

3.8.5 Jaringan Semantik

Pada bagian ini akan membahas tentang perancangan *form* yang akan dibuat kemudian diperlihatkan hubungan navigasi antara *form-form* tersebut dan pesan-pesan yang akan muncul pada setiap bagian form. Perancangan jaringan semantik dapat dilihat dalam Gambar 3.17

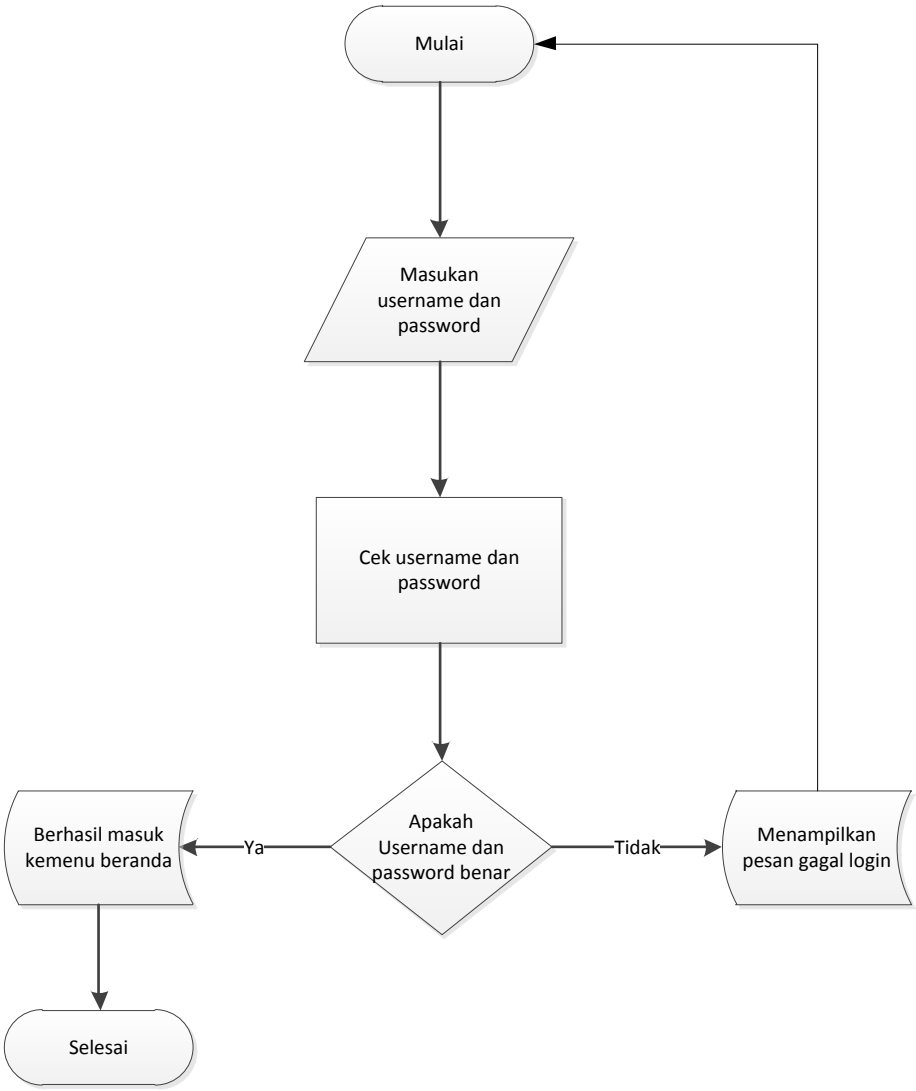


Gambar 3.17 Jaringan Semantik

3.8.6 Perancangan Prosedural

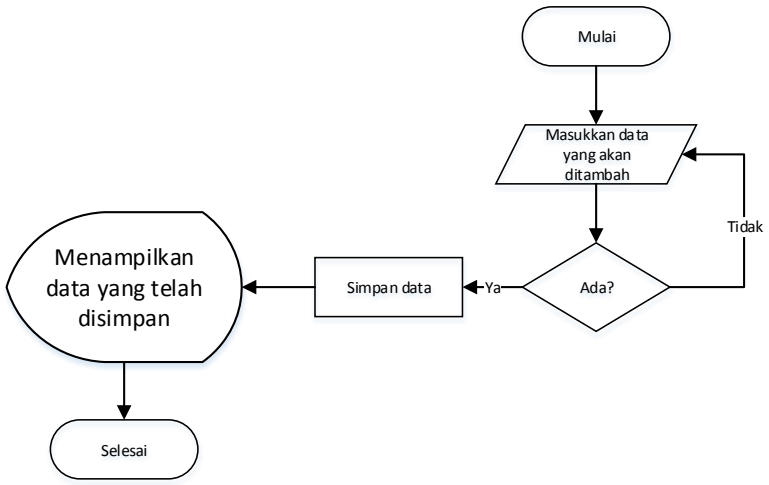
Sebagai langkah terakhir dalam perancangan yaitu perancangan prosedural yang akan diimplementasikan kedalam sistem. Prosedural ini akan digunakan sebagai algoritma dasar dalam mengkodekan prosedur yang ada.

1. Prosedur Login



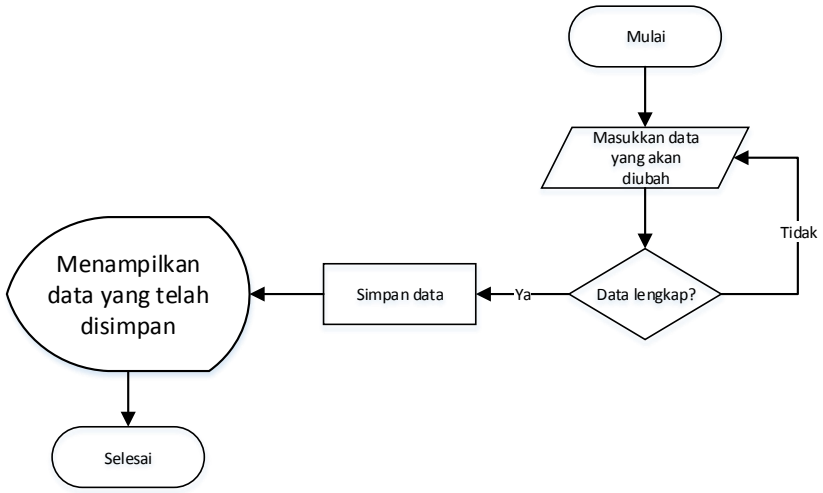
Gambar 3.18 *Flowchart* Login

2. Prosedur Tambah Data Soal



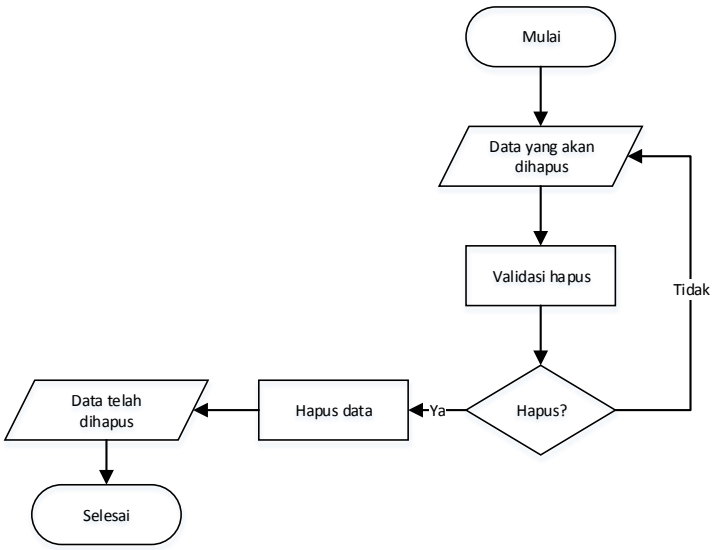
Gambar 3.19 *Flowchart* Tambah Soal

3. Prosedur Ubah Data Soal



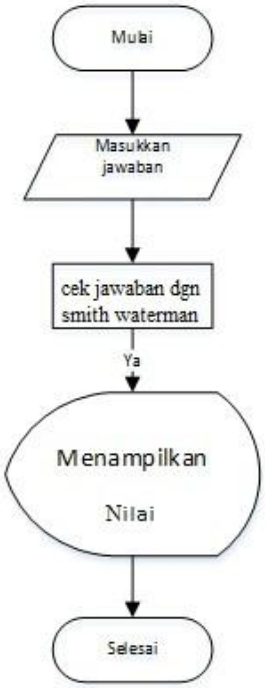
Gambar 3.20 *Flowchart* Ubah Soal

4. Prosedur Hapus Soal



Gambar 3.21 *Flowchart* Hapus Soal

5. Prosedur Mengerjakan Soal Essay



Gambar 3.26 *Flowchart* Mengerjakan Soal Essay